

Aplicació Autònoma de Picking per Veu utilitzant tecnologia MCL-Voice

Adrián Vergara Rivero

Resum– El *picking* és una activitat de preparació de comandes que consisteix en la recollida i combinació de paquets per confirmar la comanda d'un client. Aquest procés es duu a terme en els magatzems de les empreses que es dediquen a oferir serveis logístics dins la cadena de subministrament. L'objectiu d'aquest treball és dissenyar i desenvolupar una aplicació que permeti simular una activitat de picking, utilitzant un software anomenat MCL-Designer V4 i el seu mòdul MCL-Voice que permet el reconeixement de veu. Aquest projecte ha col·laborat amb la *Software Factory* del grup ZETES MULTICOM, dedicat a la investigació tecnològica en un marc de desenvolupament dels diferents processos de la cadena de subministrament.

Paraules clau– Aplicació, disseny, desenvolupament, picking, MCL-Designer V4, MCL-Voice, autònoma

Abstract– Picking is an order preparation activity that consists in pick up and combine parcels to confirm a client order. This process is carried out in company's warehouses that are dedicated to providing logistics services of the supply chain. The objective of this work is designing and developing an application that tries to simulate a picking activity, using a software named MCL-Designer V4 and its module MCL-Voice that recognizes voices inputs. This project have collaborated with ZETES MULTICOM's Software Factory, dedicated to technological research of development in the different processes of the supply chain.

Keywords– Application, design, development, picking, MCL-Designer V4, MCL-Voice, standalone

1 INTRODUCCIÓ

EL grup ZETES MULTICOM¹ té com a un dels seus propòsits principals optimitzar la cadena de subministrament amb solucions adaptades a la indústria dels seus clients i, entre d'altres, es dedica a oferir serveis logístics. Aquest projecte neix gràcies a la *Software Factory* de ZETES, situada a l'edifici Eureka de la Universitat Autònoma de Barcelona². El seu objectiu és unificar diferents idees i experiències anteriors amb prototips logístics per desenvolupar una aplicació de *picking*³ de zero que ajudi a l'empresa a créixer, enfocant-se en l'optimització del procés logístic de recollida del material.

El *picking* consisteix en la preparació de les comandes fetes pels clients, i s'encarrega de guiar a l'operari pel magatzem fins a les localitzacions dels productes i verificar que la quantitat de productes que s'agafen és la correcta.

El *picking*, depèn de l'estructura del magatzem on es duu a terme i, per tant, una aplicació de *picking* que utilitzi una empresa logística pot ser bastant diferent de la que s'utilitzi en una altra empresa del mateix sector. Per exemple, una companyia pot tenir estructurat el seu magatzem en seccions en comptes de passadissos, o tenir els productes etiquetats amb un codi de barres, o mitjançant un codi numèric llegible, etc. Per tant, els requisits d'una aplicació de *picking* venen donades per l'estructura del magatzem que utilitzi el client.

Normalment el procés de *picking* és el següent: Els operaris del magatzem tenen comandes per fer. Cada comanda assignada a un operari, consta de varis productes (*ítems*) amb les seves unitats (línies de la comanda) i, per cada ítem que l'operari ha de recollir, el sistema de *picking* guia l'operari pel magatzem fins al producte a recollir. En aquesta aplicació, l'estructura del magatzem (Fig. 10) ve

• E-mail de contacte: adriveroca@gmail.com
• Menció realitzada: Enginyeria del Software
• Treball tutoritzat per: Debora Gil Resina (Ciències de la Computació)

• Curs 2018/19

¹<https://www.zetes.com/es>

²<http://www.uab.cat/>

³<https://es.wikipedia.org/wiki/Picking>

definida per 3 coordenades, una que li indica el passadís, una altra, la prestatgeria i la tercera pel contenidor on es troba el producte. Un cop el treballador es troba davant del contenidor que li pertoca, el sistema comprova que realment es troba allà, i li demana introduir el codi de producte que es troba en el contenidor per verificar que és correcte. Depenent del magatzem, pot escanejar-lo, introduir-lo manualment i/o dir-lo en veu alta per tal de qu el sistema verifiqui que es correcte. Un cop feta aquesta verificació, el sistema passa a la següent línia de la comanda.

En aquest projecte, s'ha optat per utilitzar un sistema de reconeixement de veu per interactuar amb l'aplicació de *picking*. En concret, es farà servir el mòdul *MCL-Voice*⁴. Aquest mòdul permetrà al sistema reconèixer les respostes verbals de l'operari, cosa que un cop s'entén la lògica de negoci i la feina que realitza l'operari als magatzems, resulta molt pràctica. La possibilitat de gestionar un *picking* per veu implica haver de desenvolupar una aplicació el més còmode i intuïtiva. L'usuari final serà un treballador que voldrà fer la seva feina ràpid i sense haver d'estar agafant el terminal a cada estona per escriure dades o respondre verbalment. Com a conseqüència, el treball a desenvolupar es realitza de manera ràpida i eficaç.



Fig. 1: Picking en magatzem. Obtingut en <https://www.zonalogistica.com/el-picking-problema-o-solucion/>

1.1 Guia del document

Aquest document comença amb una breu introducció al procés logístic del *picking* i el motiu d'aquest TFG. A continuació, a l'apartat d'objectius es documenten els principals propòsits que s'han volgut assolir a l'hora de realitzar l'aplicació logística. A continuació, s'exposen els conceptes i idees més rellevants del context del *picking* a la secció de l'estat de l'art, seguida de la metodologia, on es descriu el conjunt de mètodes que s'han utilitzat durant el desenvolupament del projecte. Després, s'explica la planificació que s'ha seguit, descomponent la feina realitzada de manera ordenada i repartida en activitats. Seguidament, s'expliquen les fases de les quals s'ha compost la realització d'aquest

TFG, descrivint detalladament els diferents processos. Finalment, s'exposen els resultats obtinguts en les diferents fases del projecte i s'acaba amb les conclusions.

2 OBJECTIUS

L'objectiu d'aquest projecte és dissenyar i desenvolupar de zero una aplicació mòbil que permeti simular una activitat de *picking*, utilitzant un software anomenat *MCL-Designer V4* i utilitzant el mòdul *MCL-Voice* que permet el reconeixement verbal.

L'objectiu inicial del projecte era desenvolupar l'aplicació primer en mode local, de manera que només es treballés amb una base de dades local, i posteriorment, adaptar l'aplicació, a un mode remot que realitzés crides a un servidor. Per fer l'aplicació en remot, es requeria adaptar l'aplicació local a un model predeterminat segons un gran projecte logístic de l'empresa, *MEDEA*⁵, i realitzar una sèrie de crides predefinides al servidor. Fent proves amb aquest servidor, s'han tingut problemes per poder fer algunes de les crides que es requeria. Aquests problemes en les crides, han impedit que es pugui realitzar el desenvolupament de la part remota en la seva totalitat. Malgrat això, altres crides no han causat cap dificultat i s'ha pogut mantenir la idea de treballar amb dades remotes.

Per tant, es va decidir redimensionar alguns dels objectius que afecten només a la part remota. En comptes d'adaptar la part remota a tota l'aplicació, l'objectiu és demostrar que s'ha pogut adaptar la part local per gestionar una sèrie de comunicacions via *WebService* amb el servidor. Els objectius d'aquest projecte doncs són:

Objectiu Local:

- **OBJ1:** Dissenyar i desenvolupar una aplicació mòbil de *picking* multiplataforma i local que pugui executar-se tant en dispositius amb Sistemes Operatius Android com Windows.

Objectius remots:

- **OBJ2:** Adaptar l'aplicació per poder gestionar una comunicació amb el servidor mitjançant crides *WebService*.
- **OBJ3:** Implementar la descàrrega de la tasca de preparació de comanda al terminal amb crides al servidor

3 ESTAT DE L'ART

Dins de la cadena de subministrament s'hi troben diferents processos entre els quals; la producció de producte, operacions logístiques, l'entrega del producte al consumidor, etc. El *picking* es troba dins del procés logístic i afecta directament i en gran mesura a la productivitat de tota la cadena logística, ja que, en molts casos, és el coll d'ampolla de la d'aquesta. És un procés intensiu en mà d'obra i la seva optimització i mecanització és una de les formes de millorar el rendiment de la cadena de logística interna de les

⁴<http://www.mcl-voice.com/>

⁵<https://www.zetes.com/es/logistica-de-almacenamiento>

empreses. La millora d'aquest rendiment exigeix eliminar les parts menys productives del procés, especialment, el desplaçament entre les diferents ubicacions per on l'operari s'ha de moure per agafar els diferents productes. És per això que les empreses més importants que emmagatzemen productes, tals com Unilever, Inditex, H&M, L'Oréal o Adidas entre d'altres, tenen molt interès a millorar els seus processos de *picking*. Altres empreses tals com ZETES, Mecalex, Quonext o Ulma entre d'altres, es dediquen a proporcionar solucions software que millorin els números dels seus clients.

L'esforç per optimitzar els processos logístics ha anat evolucionant les aplicacions de *picking*, fent-les cada cop més complexes, incorporant més funcionalitats com ara:

- Que el sistema ordeni automàticament les línies a recollir i faci una combinació eficient de càrregues en funció del pes dels productes a recollir, de manera que els productes més pesats es mostrin primer. Així, es col·locaran primer en el carro que porta l'operari i els més lleugers quedaran a dalt. També en funció del volum que ocupin els productes, i no mostri els que no càpiguen al carro en funció de l'espai restant.
- Que el sistema detecti automàticament si el magatzem es queda sense *stock* d'algún producte i s'efectuï doncs, una compra per fer una reposició d'existències.
- Que el sistema s'encarregui de decidir automàticament el destí del carro un cop preparat i ple de productes. Per exemple, a quin moll de càrrega enviar-lo.
- Que el sistema calculi una ruta eficient pel magatzem per no tenir a l'operari donant voltes i passar varies vegades pel mateix lloc.

Òbviament totes aquestes funcionalitats no es troben en totes les aplicacions de *picking*, sinó que depenen de l'estructura del magatzem del client i del preu que es vulgui pagar per l'aplicació.

4 METODOLOGIA

Per realitzar aquest projecte i totes les diferents fases d'aquest, s'ha optat per utilitzar la metodologia d'Enginyeria del Software: àgil⁶. Bàsicament es basa en l'adaptabilitat de qualsevol canvi del projecte com a mitjà per augmentar les possibilitats d'èxit d'un projecte, fent de forma iterativa els següents processos: Planificació, desenvolupament, execució i test. La seva utilització ha permès fer profit de les següents capacitats:

Augmenta les possibilitats d'èxit del projecte, intentant minimitzar el risc desenvolupant en iteracions. A més, divideix el projecte principal en diferents subprojectes amb l'objectiu de reduir la dificultat de desenvolupament en cadascun d'ells. D'aquesta manera és més fàcil adaptar qualsevol canvi d'implementació havent només de modificar un o alguns dels subprojectes i no haver de tractar el canvi en el context d'un projecte general gran que després podria propagar alteracions en el codi d'altres subprojectes

⁶<http://agilemethodology.org/>

modificant la funcionalitat. També s'adapta als canvis de requeriments encara que el desenvolupament estigui en una fase avançada, utilitzant aquests canvis com a avantatge competitiu pels clients. Per acabar, permet realitzar la fase de *testing* de forma paral·lela amb el desenvolupament del codi. Això resulta especialment útil en aquest projecte, ja que cada cop que es fa qualsevol petita implementació es pot provar i validar aquesta petita peça del programari.

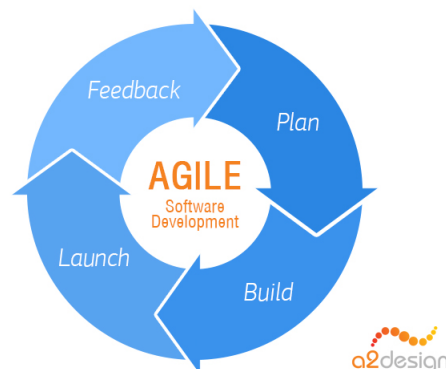


Fig. 2: Metodologia àgil. Obtingut en <http://www.a2design.biz/blog/why-should-you-use-agile-software-development-methodology/>

5 PLANIFICACIÓ

Per fer aquest TFG, s'han definit de manera ordenada una sèrie d'activitats i subactivitats a desenvolupar (Fig. 9), per tal d'assolir l'objectiu de realitzar un procés d'enginyeria del software. La definició d'aquestes tasques, ha permès tenir un control de la feina feta i per fer en tot moment durant la realització del treball.

5.1 Fases i activitats

Fase 1. Formació i investigació: Familiaritzar-se amb l'entorn de treball de l'empresa, *MCL-Designer V4*, el mòdul *MCL-Voice*, fer un estudi de projectes i idees anteriors de *pickings* de l'empresa i visitar als magatzems de Zetes situats a polinyà per entendre millor el negoci.

Fase 2. Disseny: Elaboració dels requeriments de l'aplicació, casos d'ús, planificar el projecte, fer el diagrama de flux, el prototip o interfície d'usuari i la base de dades de l'aplicació.

Fase 3. Desenvolupament local: Elaboració de tota la aplicació de *picking* de zero, utilitzant *MCL-Designer V4* i el seu mòdul *MCL-Voice*, desenvolupant la interfície d'usuari i tota la seva lògica.

Fase 4. Desenvolupament remot: Adaptació de l'aplicació local al projecte existent de l'empresa, *MEDEA*, de manera que l'aplicació realitzi crides a un servidor i treballi amb dades remotes. Havent d'adaptar l'aplicació local al model predeterminat segons el model utilitzat a *MEDEA* i realitzar una sèrie de crides predefinides a un servidor.

Fase 5. Test: Comprovar el correcte funcionament del disseny i dels requisits documentats i corregir errors.

Cal destacar que tant el desenvolupament de l'aplicació com el testeig s'han fet de manera simultània al desenvolupament, gràcies a la metodologia àgil que s'ha fet servir.

Fase	Data
Formació i Investigació	19/02/2018 - 11/03/2018
Disseny	12/03/2018 - 14/03/2018
Desenvolupament local	15/03/2018 - 23/04/2018
Desenvolupament remot	24/04/2018 - 18/05/2018
Test	15/03/2018 - 18/05/2018

TAULA 1: FASES DEL PROJECTE SIMPLIFICADES

5.2 eines

Per poder realitzar aquest projecte, s'han utilitzat els següents softwares i dispositius;

5.2.1 Software MCL

- **MCL-Designer V4:** Plataforma que permet desenvolupar aplicacions mòbils de manera ràpida i intuïtiva. Pot: Compilar (*build*) aplicacions mòbils per a diferents Sistemes Operatius com Windows (WinCE, Win-MOB) o Android. Desplegar (*deploy*) aplicacions i software de manera ràpida i eficient. Executar (*run*) aplicacions de manera optimitzada.
- **MCL-Voice:** Mòdul de MCL amb el qual serà possible programar ordres del sistema cap a l'usuari i per altra banda, fer el reconeixement vocal per la confirmació de les ordres rebudes.
Fa possible que l'operari interactuï amb el sistema, parlant a través d'un micròfon que anirà connectat al mateix dispositiu.

5.2.2 Software de gestió de projectes

S'han utilitzat diferents softwares per dissenyar i elaborar diagrames o fer la planificació;

- **Draw.io:** Software online que permet crear diagrames de flux de manera gratuïta.
- **Fluidui.com:** Software online que permet crear prototips d'aplicacions mòbils o de web de manera ràpida i gratuïta.
- **Microsoft Project:** Software que permet l'assignació de recursos a tasques, donar seguiment al progrés, administrar pressupost i analitzar càrregues de treball.

5.2.3 Dispositius mòbils Toughpads

MCL-Designer V4 permet executar les aplicacions en dispositius mòbils (*Toughpads*) de múltiples fabricants diferents amb sistema operatiu Android o Windows, entre ells: ATID, Datalogic, Honeywell, Intermec, Motorola/Symbol/Zebra, Panasonic, Samsung i Unitech. Amb aquests terminals els operaris podran realitzar el procés de recollida de material, que també ens han permès testear l'aplicació a mesura

que el projecte avançava. Per realitzar aquest projecte, s'han utilitzat els dispositius *Panasonic Toughpad FZ-N1*⁷ i *Zebra WT6000*⁸ per testear l'aplicació.



Fig. 3: MCL-Voice. Obtingut en http://www.mcl-collection.com/products/mcl_voice.php

6 DESENVOLUPAMENT DEL TFG

A continuació es defineix en què ha consistit cadascuna de les fases del projecte; Formació i investigació, disseny, desenvolupament local, desenvolupament remot i testeig:

6.1 Formació i investigació

La fase de formació i investigació va incloure les següents tasques:

6.1.1 Familiarització amb l'entorn de treball

Les primeres setmanes del projecte van servir per realitzar una primera presa de contacte amb l'empresa, coneixent les tecnologies que utilitzen i visitant els magatzems de ZETES ubicats a Polinyà on es va poder conèixer l'estructura real d'un magatzem i quina és la forma de treballar d'un operari al realitzar un servei logístic de *picking*.

6.1.2 Formació en MCL-Designer V4

També es van aprofitar els primers dies de l'estada per familiaritzar-se amb el software amb el qual s'ha realitzat l'aplicació, *MCL-Designer V4*. Es va aconseguir adquirir el coneixement per utilitzar el programa, primerament, gràcies a l'ajuda d'un parell de companys que van explicar els conceptes principals del software. Després, mitjançant l'assistència a un *meeting* que es va dur a terme a l'hotel campus UAB on vaig aprendre a utilitzar correctament el mòdul *MCL-Voice*.

6.1.3 Estudi d'idees i experiències anteriors

També es va aprofitar aquesta primera etapa per analitzar i entendre l'aplicació a nivell de codi el gran projecte logístic de l'empresa, el MEDEA. És el gran projecte logístic de

⁷<https://na.panasonic.com/us/computers-tablets-handhelds/handhelds/handhelds/toughpad-fz-n1>

⁸<https://www.zebra.com/es/es/products/mobile-computers/wearable-computers/zt6000.html>

ZETES, que aconsegueix que el personal del magatzem realitzi les seves tasques amb major rapidesa i exactitud, des de la recepció fins a l'enviament del producte.

6.2 disseny

En aquesta fase s'ha elaborat el disseny de documents que shan utilitzat per definir les diferents activitats en les quals s'ha descompost el projecte. S'ha dissenyat els requisits, casos d'ús, diagrama de flux, base de dades i prototip de l'interfície d'usuari.

6.2.1 Requisits

S'ha recollit un conjunt de descripcions que defineixen el comportament de l'aplicació agrupats en requisits:

Requisits Funcionals	
Requisits	Descripció
RF-01	El sistema ha d'acceptar respostes verbals i manuals amb la interfície, és a dir, ser multimodal (OBJ1)
RF-02	El sistema ha d'utilitzar la tecnologia del mòdul MCL-Voice (OBJ1)
RF-03	El sistema ha d'identificar a l'usuari mitjançant un codi d'usuari únicament (OBJ1)
RF-04	El sistema ha de descarregar la tasca de preparació de comanda que el treballador seleccioni (OBJ1) y (OBJ3)
RF-05	El sistema ha de guiar a l'usuari pel magatzem fins a la localització del producte (OBJ1)
RF-06	El sistema ha de guardar la localització d'un producte donats 3 números; un pel passadís del magatzem, un altre per l'estanteria i un tercer per definir el contenidor de prestatgeria on es troba (OBJ1)
RF-07	A l'hora de validar la localització, l'usuari pot escanejar un codi de barres que hi ha en el contenidor on es troba el producte (OBJ1)
RF-08	El sistema ha de validar les quantitats correctament preparades sobre les que inicialment havia de preparar (OBJ1)
RF-09	El sistema ha d'actualitzar l'estat d'una línia a "completed" o similar, quan es completi la recollida del producte de la mateixa (OBJ1)
RF-10	El sistema ha de permetre a l'usuari saltar-se una línia en el moment que digui "skip line", no actualitzar el seu estat i guardar-la per la següent iteració, on només es demanarà les línies no completes (OBJ1)

6.2.2 Casos d'ús

S'ha elaborat un diagrama de casos d'ús per descriure el comportament que té el sistema mitjançant la interacció amb l'usuari. A partir d'aquest diagrama es van extreure part dels requisits de l'aplicació.

6.2.3 Diagrama de flux

S'ha elaborat una representació gràfica de l'algorisme que executa l'aplicació (Fig. 8). Representa el flux d'execució mitjançant fletxes que connecten els punts d'inici i de finalització. L'aplicació comença gestionant el login d'usuari, demanant el seu codi i a continuació

6.2.4 Base de dades

També s'ha fet el disseny de la base de dades que utilitza l'aplicació per diferents processos com; fer el *Login* dels usuaris, carregar les comandes i línies assignades als usuaris, actualitzar l'estat de les línies, portar un control dels productes recollits i que falten per recollir, etc.

PickingID	UserCode	TotalLines	StatusHeader
1	1	2	0
2	1	1	0
3	2	1	0
4	3	4	0
0	0	6	0
6	4	1	0
7	4	1	0
8	4	1	0

Fig. 4: Taula de les comandes de la base de dades

6.2.5 Prototip o interfície d'usuari

S'ha considerat pertinent realitzar un prototip de la interfície d'usuari de l'aplicació abans de començar a fer el propi desenvolupament d'aquesta. En ser el producte final una aplicació mòbil, ha sigut molt útil desenvolupar un prototip que permetís visualitzar i tenir clara l'estructura de les diferents pantalles de l'aplicació abans de començar a desenvolupar-les i simular tota la navegació de l'aplicació entre aquestes. L'estètica s'ha conservat en gran mesura respecte a l'aplicació final.

6.3 Desenvolupament del mode local

Primerament s'ha desenvolupat el mode local, en el qual el sistema conté totes les dades carregades en una base de dades local. El terminal doncs, conté tota la lògica necessària per gestionar el procés logístic.

6.3.1 Picking standalone

L'aplicació resultant és un *picking* autònom, això significa que el terminal té programada tota la lògica i que no necessita cap servidor, host o ERP per dur a terme el procés logístic.

6.3.2 Aplicació multimodal

L'aplicació també és multimodal, el que significa que l'operari pot confirmar, entrar dades o navegar entre les diferents pantalles de diferents maneres a la vegada. És a dir, pot confirmar per exemple el codi de localització de producte de manera oral, manual o mitjançant l'*scanner* d'un codi de barres. El sistema no pregunta en cap moment a l'usuari, per quina d'aquestes opcions es decanta, sinó que des d'un primer moment està preparat per rebre qualsevol dels tres tipus d'inputs i mentre està escoltant per si rep una ordre verbal, alhora està preparat per si rep qualsevol de les dues

altres possibles entrades. Això és possible gràcies al mòdul *MCL-Voice*.

6.3.3 Procediments Globals i Locals

L'aplicació s'ha estructurat en procediments globals, els quals contenen tot el codi de l'aplicació, repartit en rutines de les diferents pantalles. També s'han creat diferents procediments locals o funcions, inclosos dins dels globals, que engloben una part de codi que té una funció en concret i és propensa a executar-se varies vegades desde diferents punts de l'aplicació. Els principals procediments de l'aplicació són:

1. Login: Aquest procediment global s'encarrega de reconèixer quin usuari dels de la base de dades s'està identificant. Conté el següent procediment local:

- **ReadUserNameFromDataBase:** S'encarrega d'informar al sistema del nom de l'usuari, si existeix, el qual s'identifica mitjançant un codi.

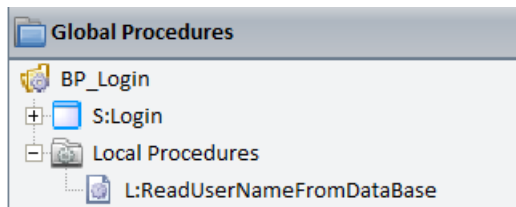


Fig. 5: Procediment global Login i els seus procediments locals

2. Picking: Aquest procediment global és el que gestiona tots els processos relacionats amb el procés logístic del *picking*. Conté els següents procediments locals:

- **ProcessHeaderPicked:** S'encarrega de fer saber al sistema quina comanda ha escollit l'usuari, sigui per veu o escollint-la manualment.
- **SkipLine:** S'encarrega de gestionar el salt de línia. Quan l'usuari decideixi saltar la línia que estigui fent, aquest procediment local procedirà a mostrar la següent línia de la comanda actual i guardar la línia saltada per la següent iteració, on es demanarà fer les línies que faltin per fer.
- **TryJumpLocation:** S'encarrega de comprovar automàticament si la línia anterior que ha realitzat l'operari i l'actual, comparteixen part de la localització, de manera que si les dues línies es troben al mateix passadís, el sistema no informarà del passadís on es troba el producte, sinó que directament informarà de la prestatgeria i/o el contenidor.
- **CountHowManyLinesAreCompleted:** S'encarrega d'informar al sistema cada cop que es completa una línia, de quantes línies de la comanda actual estan completades, de manera que es podrà saber quan s'hagin completat totes les línies de la comanda.
- **TryConfirmPickLine:** S'encarrega d'informar al sistema de, un cop agafant el producte, si l'usuari ha agafat la quantitat demanada de producte, n'ha agafat menys o està intentant agafar-ne de més.

- **GetLineInfo:** S'encarrega d'obtenir tota la informació de la línia actual de la base de dades. Així com a quina comanda pertany, la localització on es troba el producte, la quantitat de producte per agafar, la quantitat que porta agafada, etc.
- **GetProductInfo:** S'encarrega d'obtenir tota la informació d'un producte donat. Així com la descripció, el preu, etc.
- **CopyLinesToAuxLines:** S'encarrega de copiar la informació de les línies de la comanda actual a un fitxer auxiliar que guarda la informació de si per la iteració actual de les línies d'una comanda, s'han consultat totes elles.
- **GetLineInfoFromAuxLines:** S'encarrega d'obtenir tota la informació de les línies de la comanda actual del fitxer auxiliar.
- **ResetExploredAuxLines:** S'encarrega de netejar un camp del fitxer auxiliar que guarda la informació de si per la iteració actual de les línies d'una comanda, s'han consultat totes elles. De manera que quan s'hagin consultat, es neteja aquest camp i a la següent iteració es demanarà fer les línies que faltin per fer.

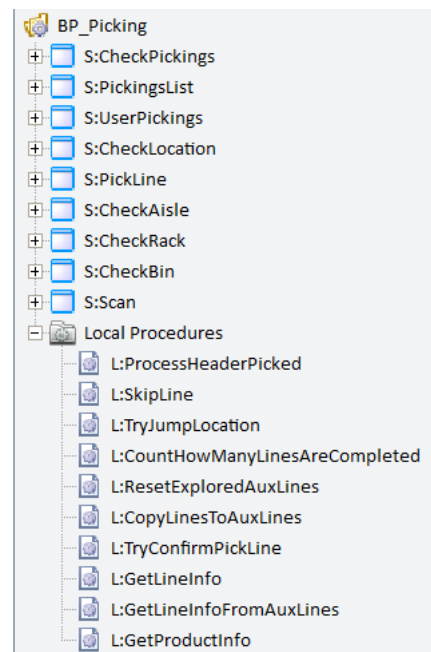


Fig. 6: Procediment global Picking i els seus procediments locals

També s'ha utilitzat un altre procediment global o llibreria, ja implementat però, reutilitzat del projecte MEDEA.

Language: Aquesta llibreria del projecte MEDEA fa possible el canvi d'idioma. Té el següent procediment local:

- **Create_Languages_List:** S'encarrega d'informar al sistema del canvi de llengua, tant de l'idioma amb el qual l'operari s'ha de comunicar amb el terminal, com de l'idioma amb el que terminal mostra la informació al treballador.

6.4 Desenvolupament del mode remot

Com s'ha comentat a la secció dels objectius, la idea principal era agafar la mateixa aplicació local feta anteriorment i adaptar-la al model del projecte MEDEA. De manera que treballés amb dades remotes mitjançant un servidor de l'empresa i realitzés la mateixa lògica ja implementada, però en remot. La idea era doncs reutilitzar les crides als servidors de MEDEA i comunicar-se amb aquest servidor.

Tanmateix, fent proves, s'han trobat problemes per fer algunes de les crides que es requeria. Només s'ha produït en algunes d'elles, que han impedit que es pugui realitzar el desenvolupament remot en la seva totalitat. Altres, però, no han causat cap dificultat i s'ha pogut mantenir la idea de treballar amb dades remotes.

De manera que s'han pogut gestionar de manera remota les següents funcionalitats;

- Login: El servidor retorna si l'usuari que s'està identificant existeix a la base de dades remota i si té permisos de veu per poder utilitzar l'aplicació.
- El terminal descarrega la tasca de preparació de comanda amb crides al servidor.

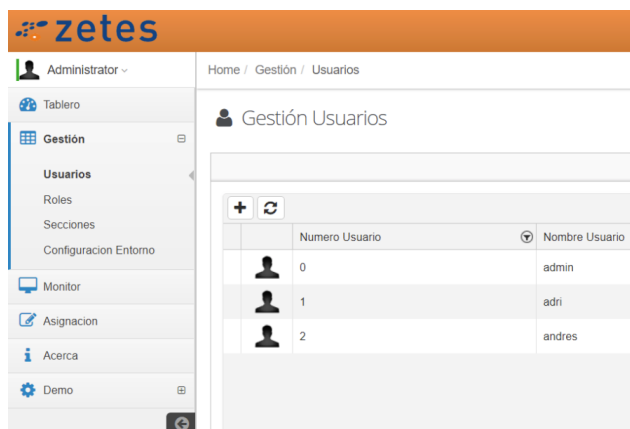


Fig. 7: Dashboard MEDEA

6.5 Testeig

S'ha comprovat que el funcionament de l'aplicació final és correcte i compleix els requisits funcionals contemplats. Pel que fa al desenvolupament, el mode local s'ha estat testejant constantment tant en windows, *debugant* amb el propi software de MCL, com en Android, amb els dispositius *toughpad*. S'ha creat tota una base de dades fictícia d'usuaris amb comandes assignades, línies i productes. D'aquesta manera es podia comprobar si el comportament de l'aplicació era l'esperat i seguir tots els fluxos de les dades que són processades pel sistema i identificar i corregir errors de codi que puguin desencadenar resultats indesitjats. De la mateixa manera, també s'ha testejat tota la funcionalitat per veure si el sistema responia de la manera esperada amb la veu.

7 PRESENTACIÓ DE RESULTATS

L'aplicació té una pantalla inicial (Fig. 11) on l'usuari pot introduir el seu codi d'usuari manualment o bé dir-lo en veu alta + una confirmació (input de veu: <número> 'ready' / <número> 'OK'). També pot canviar l'idioma de l'aplicació (Fig. 12) al castellà, que per defecte està en anglès. Quan es canvia l'idioma, es canvia l'idioma amb el qual el sistema es comunica amb l'usuari i també amb el qual l'usuari ho ha de fer amb el sistema, gràcies al procediment global L-Language"anomenat al punt 6.3.3.

Un cop l'usuari s'ha identificat, se li mostra una pantalla (Fig. 13) on pot consultar les seves tasques a realitzar, segons la feina que tingui assignada a la base de dades (Fig. 14), o pel contrari, realitzar feina que no estigui assignada a cap treballador, la qual és accessible des de tots els usuaris. Escollirà doncs, de manera manual o verbal, quina comanda vol fer.

Si l'usuari escull la primera opció, se li mostrarà una llista (Fig. 14) amb la seva feina a fer amb el nombre de línies a realitzar que té cada tasca. Una línia està composta per un ítem (X quantitat d'un producte concret). Quan esculli quina tasca fer, es començarà a fer el procés de *picking* per la primera línia de la tasca escollida. Se li mostrarà a quina localització (Fig. 10) es troba el producte dins del magatzem. Una localització que ve donada segons tres coordenades; Una primera (Fig. 15) que indica a quin passadís es troba, una segona que indica la prestatgeria i una tercera i última que representa el contenidor i localització exacta del producte. Cada localització té la seva pròpia pantalla dins l'aplicació i són gairebé iguals. L'usuari anirà confirmant (input de veu: 'ready' / OK) en cadascuna d'aquestes pantalles per accedir a la següent.

Un cop l'usuari es trobi davant del producte, se li demanarà un codi de localització per comprovar que realment es troba allà (Fig. 16) i podrà confirmar-lo manualment, o per veu o bé escanejar una etiqueta, si és que el magatzem disposa d'etiquetes associades a les localitzacions dels productes. El sistema comprovarà l'input de l'usuari. En el cas de ser correcte, es procedirà a fer la recollida de producte (Fig. 17), on l'usuari haurà d'agafar X unitats d'aquest producte. Pot agafar la quantitat demanada (input de veu: <quantitat> + 'ready' / <quantitat> + 'OK') o una menor quantitat, si per exemple no hi ha suficient *stock* del demanat (input de veu: <quantitat> + 'short'), però en cap cas podrà agafar més quantitat de la demanada. En cas de tenir que agafar molta quantitat d'un producte, l'usuari anirà agafant petites quantitats mentre informa al sistema, i aquest anirà avisant a l'operari de quantes unitats li queden per agafar. Quan l'usuari reculli les quantitats de producte demanades, el sistema confirmarà aquesta línia realitzada i, en cas d'haver més línies per fer de l'actual tasca, se li mostrarà la localització de la següent línia (Fig. 15) i es tornarà a repetir el procés. En cas d'haver realitzat l'última línia de la tasca actual, s'actualitzarà l'estat de la tasca completada i es mostraran les tasques restants per fer (Fig. 14), on ja no apareix la tasca que s'acaba de completar.

L'aplicació compta amb altres funcionalitats pràctiques un cop es coneix la logística d'un magatzem, com per

exemple:

- Si la localització del producte de la línia que s'acaba de fer és semblant a la localització del producte de la línia següent, no es repeteixen les 3 pantalles de localització, sinó que només es mostren les localitzacions que són diferents respecte a l'anterior.
- L'operari pot saltar una línia en qualsevol moment. Llavors, el sistema mostra la següent línia de la tasca i quan es completin les següents, es tornaran a demanar les línies que s'hagin saltat anteriorment.

8 CONCLUSIONS

Aquest projecte ha consistit en realitzar un procés d'Enginyeria del Software, desenvolupant una aplicació autònoma, mòbil, multiplataforma de *picking* i multimodal.

Inicialment, dissenyant l'aplicació i elaborant diferents documents que es poden trobar adjunts a aquest informe, tals com el diagrama de flux de l'aplicació, el prototip o interfície d'usuari, els requeriments o la planificació. Posteriorment, amb el desenvolupament de l'aplicació, utilitzant el programa *MCL-Designer V4* i elaborant un software des de zero. Programant tota la lògica aplicant una sèrie de metodologies i bones pràctiques que han permès l'elaboració d'aquest projecte i aconseguir que el software programat sigui més complet, entenable, mantenible i fàcil de continuar desenvolupant.

Aquesta aplicació desenvolupada simula una activitat de *picking* que podria dur a terme un operari en un magatzem per realitzar la seva feina logística. El resultat de l'aplicació ha sigut l'esperat. És totalment funcional i permet simular un procés de *picking* amb múltiples funcionalitats com la validació de l'usuari, elecció de quina tasca fer, guiatge fins a la localització del producte, validació del dígit de producte, procés de realització de totes les línies de la comanda i tancament d'aquesta. En aquest projecte, tot i haver acabat tota l'aplicació processant dades en local, a més s'ha començat a adaptar-la per a gestionar una sèrie de comunicacions amb el servidor via *WebService* per a processar dades remotes, obtenint-les d'un servidor.

De cara al futur, el software encara és millorable. A partir de la base generada per l'actual, una gran part de la lògica de tot el procés del *picking* es pot aprofitar per altres projectes logístics. També, es pot seguir elaborant el propi desenvolupament de l'aplicació remota fins a donar-se per completada, si l'empresa Zetes, per la qual s'ha desenvolupat aquest projecte, ho creu oportú.

Pel que fa als objectius del TFG, no s'han pogut assolir els objectius inicials pels problemes esmentats anteriorment, però un cop redimensionats, sí que han quedat tots satisfets.

9 AGRAÏMENTS

M'agradaria començar donant les gràcies a ZETES, que és l'empresa que va fer possible que aquest projecte s'hagi pogut dur a terme. També, als meus companys de la Software

Factory de ZETES, en especial a dos d'ells, Javier Roca i Óscar Pérez. També a la professora Debora Gil, que ha estat sempre disponible i donant-me consells de millora de manera constant. Per últim, a la meua família que sempre m'ha transmès el seu suport des de l'inici del Grau.

REFERÈNCIES

- [1] 4 warehouse best practices for stellar performance (maig 2018) <https://www.benchmarkingsuccess.com/4-warehouse-best-practices-for-stellar-performance/>
- [2] Tècniques to improve the warehouse order picking process (abril 2018) Obtingut en: <https://sipmm.edu.sg/techniques-to-improve-the-warehouse-order-picking-process/>
- [3] Les principals raons per les quals escollir la metodologia àgil (març 2018) Obtingut en: <https://hygger.io/blog/the-most-remarkable-reasons-to-go-agile/>
- [4] Les 12 raons claus per les quals les empreses adopten la metodologia àgil (març 2018) Obtingut en: <https://www.leadingagile.com/2011/01/the-12-key-reasons-companies-adopt-agile/>
- [5] ZETES MEDEA. Logística d'emmagatzematge (març 2018) <https://www.zetes.com/es/logistica-de-almacenamiento>
- [6] Software, The Latex Project (febrer 2018) <https://www.latex-project.org/>
- [7] Documentació de MCL-Designer V4 (abril 2018) http://files.mcl4e.com/mcl4/help/designer/4.6/index.html?getting_started_with_mcldesignerv4.htm

Functionality Diagram

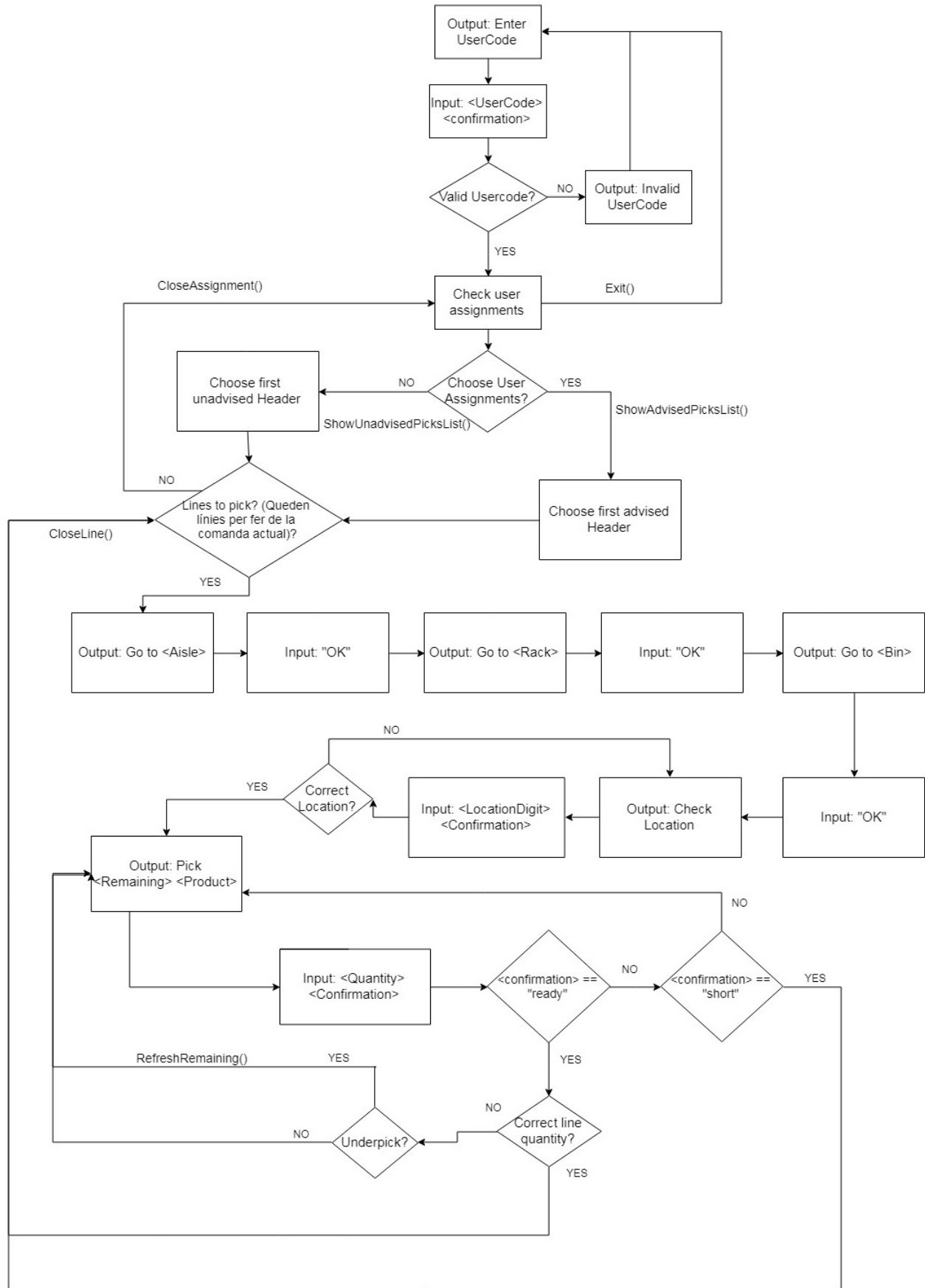


Fig. 8: Diagrama de flux de l'aplicació

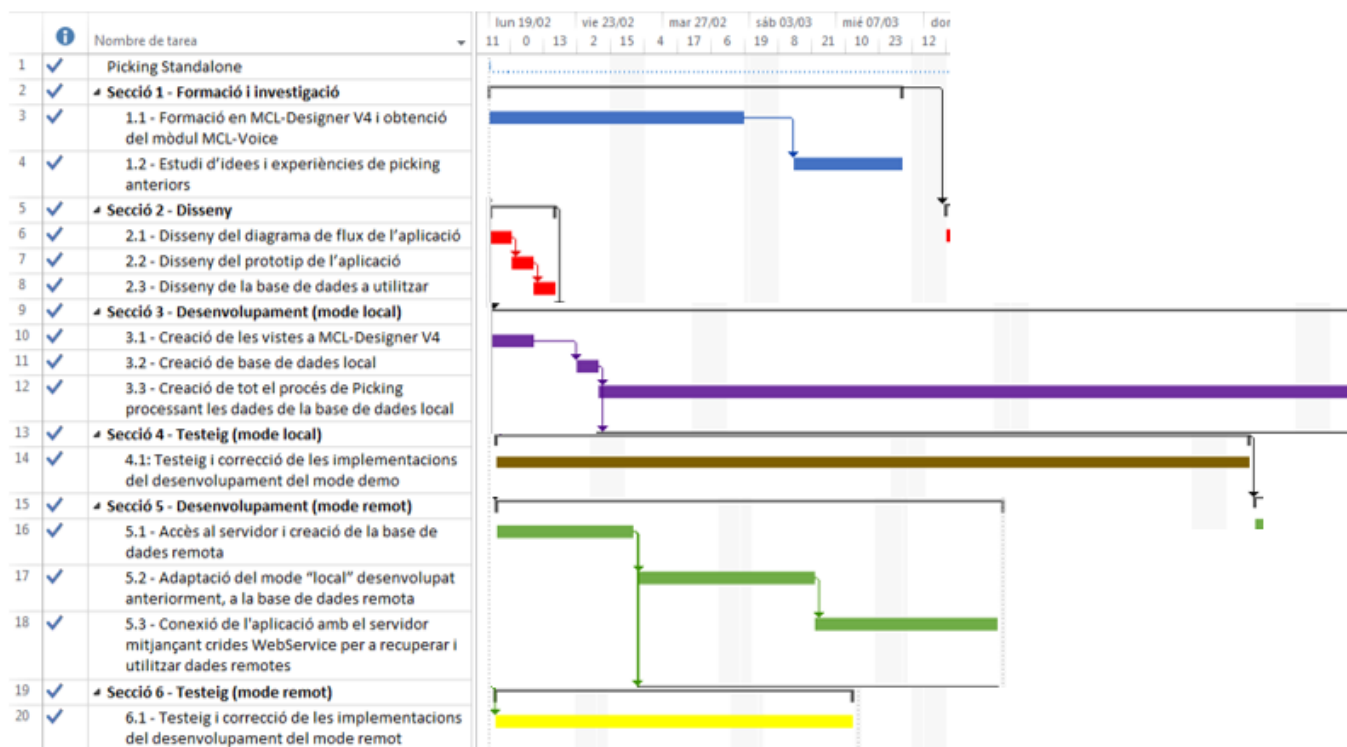


Fig. 9: Diagrama de Gantt de la planificació

Blau: Fase de Formació i Investigació Vermell: Disseny
 Púrpura: Desenvolupament del mode local Marró: Testeig del mode local
 Verd: Desenvolupament del mode remot Groc: Testeig del mode remot

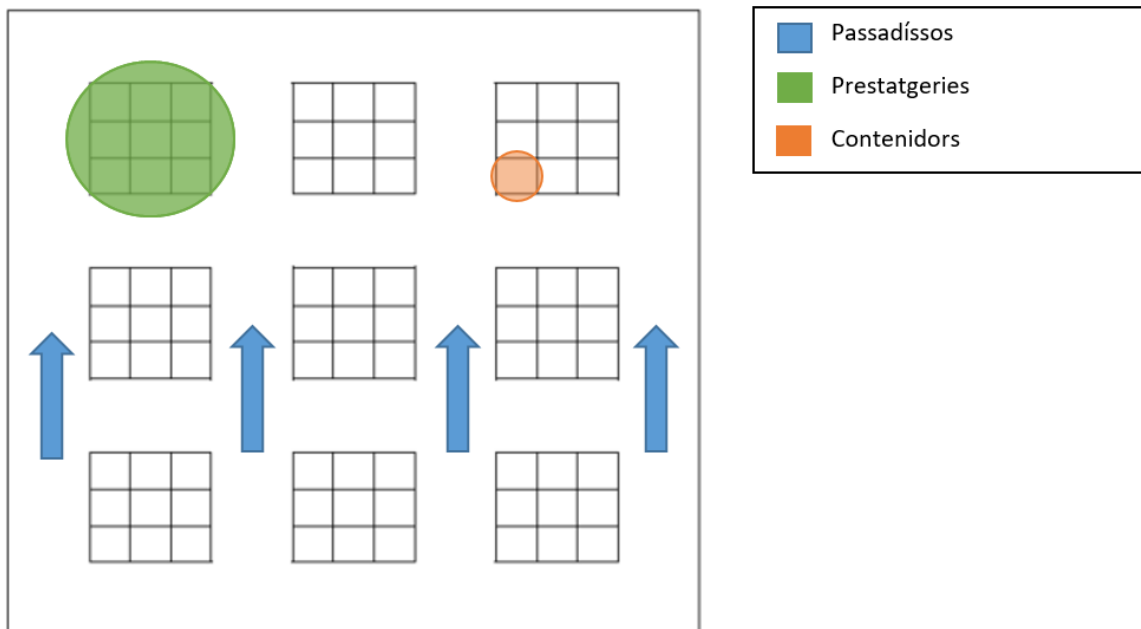


Fig. 10: Estructura de magatzem

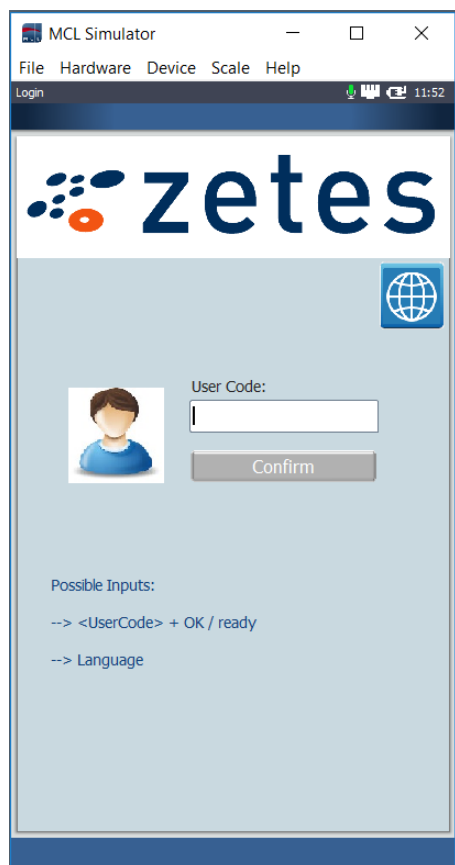


Fig. 11: Pantalla inicial d'identificació d'usuari

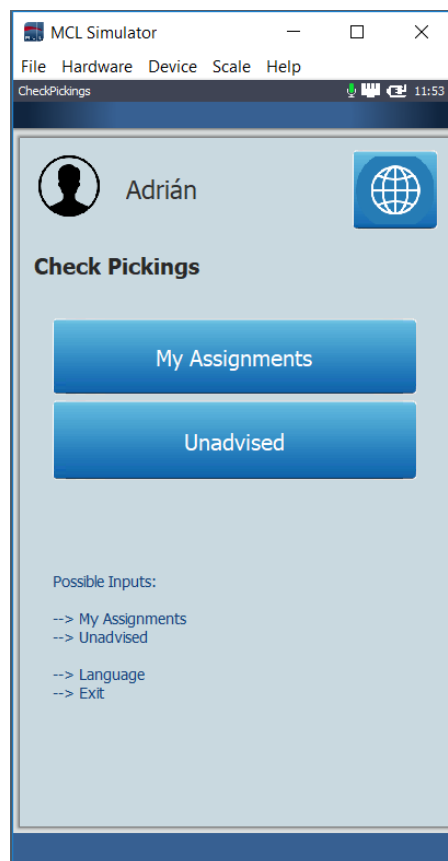


Fig. 13: Menú principal

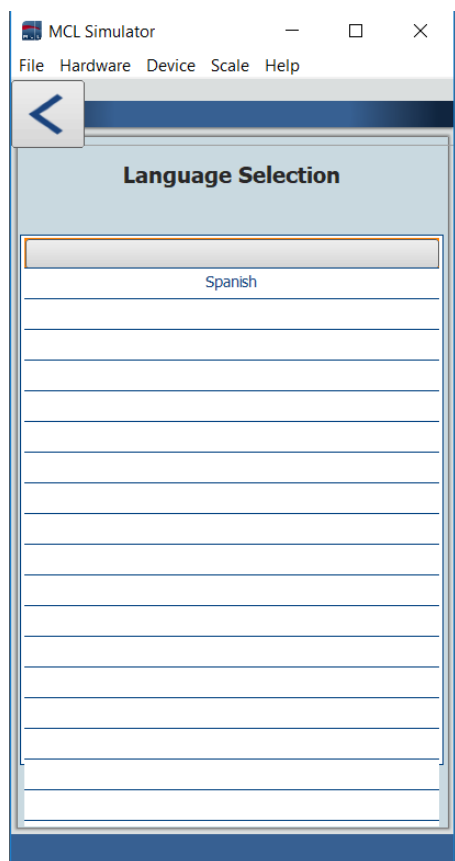


Fig. 12: Selecció d'idioma

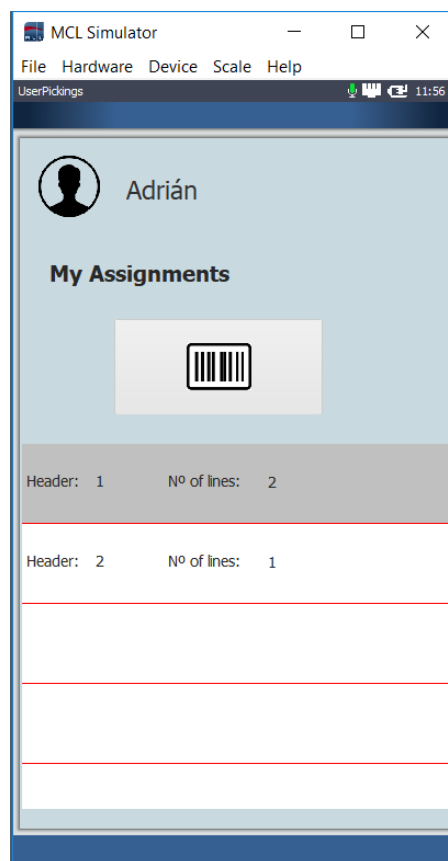


Fig. 14: Consulta de tasques assignats

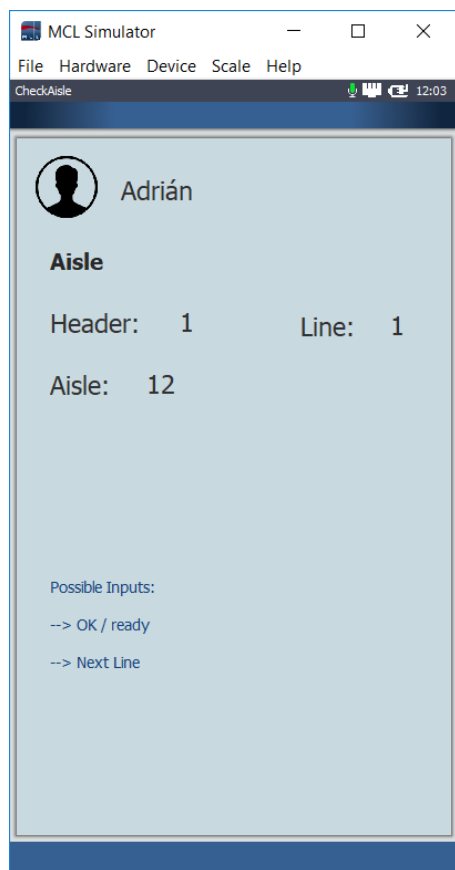


Fig. 15: Localització 1, Passadís

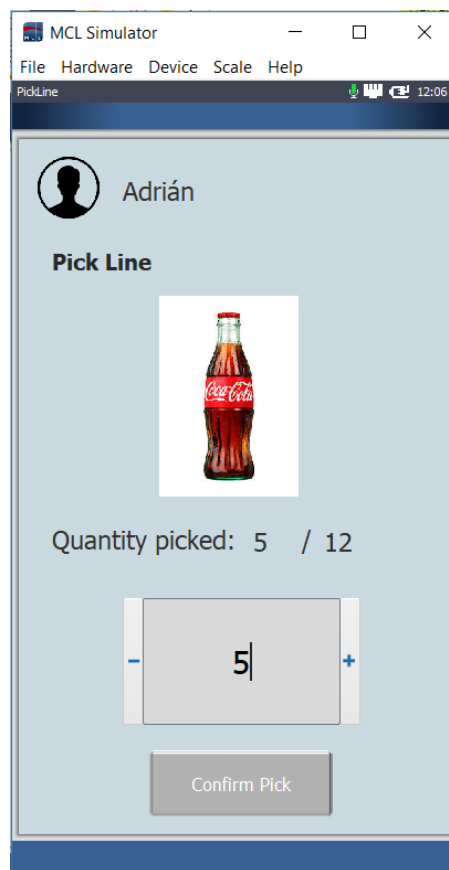


Fig. 17: Recollida de producte

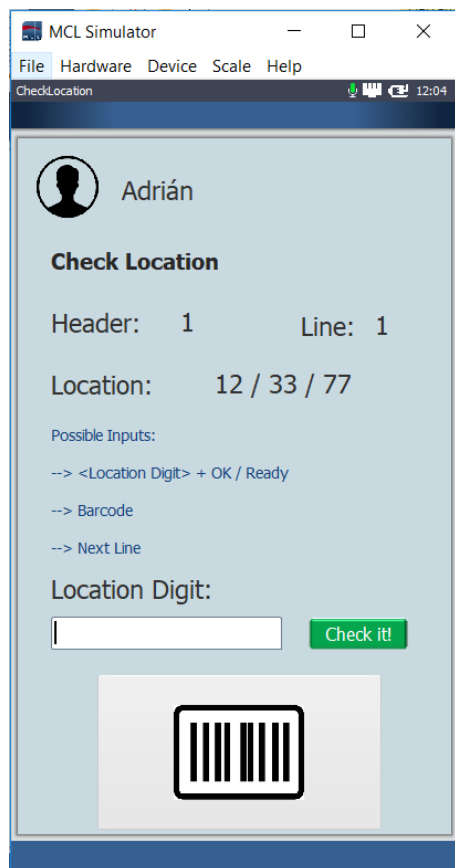


Fig. 16: Comprovació de codi de localització